

SN3262 – Network Administration, Management & Security

MIB-II, SMI and ASN.1

Reading:

MIB-II RFC 1213

SMI RFC 1155

Some useful URLs that you should look at.

[http://ironbark.bendigo.latrobe.edu.au/subjects/CN/2008/lectures/l21.d/index.html#\(1\)](http://ironbark.bendigo.latrobe.edu.au/subjects/CN/2008/lectures/l21.d/index.html#(1))

[http://ironbark.bendigo.latrobe.edu.au/subjects/CN/2008/lectures/l22.d/index.html#\(1\)](http://ironbark.bendigo.latrobe.edu.au/subjects/CN/2008/lectures/l22.d/index.html#(1))

[http://ironbark.bendigo.latrobe.edu.au/subjects/CN/2008/lectures/l23.d/index.html#\(1\)](http://ironbark.bendigo.latrobe.edu.au/subjects/CN/2008/lectures/l23.d/index.html#(1))

<http://www.et.put.poznan.pl/snmp/main/mainmenu.html>

All of the above accessed 29/10/08

What is being dealt with in this unit?

- What (from a semantic viewpoint) is being monitored? And what form of control can be exercised by the network administrator?
- What is the specific form of the information that will be reported and/or exchanged?

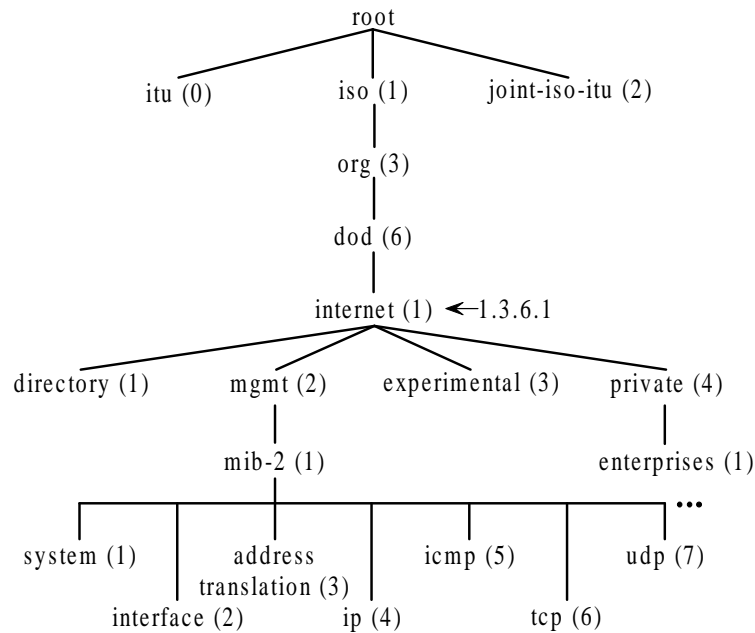
MIBs

- Management information is represented as a collection of managed objects that form a virtual information store called a Management Information Base(MIB).
- A MIB contains definitions of network management objects.
- MIB objects, e.g. a counter containing the number of IP datagrams discarded by a router due to errors in the header, define the management information maintained by a managed node.
- Related MIB objects form MIB modules.
- The **Structure of Management Information** (SMI) (rfcs 1155 and 2578 define the data types and rules for management information. Formal descriptions of the structure are given using Abstract Syntax Notation One (ASN.1) (ISO — 8824 and ISO — X.680).

SMI

The SMI is essentially a definition language that uses ASN.1 and ensures that the syntax and semantics of network-management data are well defined and unambiguous. It does not define specific instances of the data in a managed entity but the language in which such information is specified.

Each type of object (termed an object type) has a name, a syntax, and an encoding. The name is represented uniquely as an OBJECT IDENTIFIER. An OBJECT IDENTIFIER is an administratively assigned name using the ISO object identification framework in which objects are named in a hierarchial manner.



The syntax for an object type defines the abstract data structure corresponding to that object type. For example, the structure of a given object type might be an INTEGER or OCTET STRING. Only a subset of the ASN.1 are used for the sake of simplicity.

The encoding of an object type is how instances of that object type are represented using the object's type syntax. Implicitly tied to the notion of an object's syntax and encoding is how the object is represented when being transmitted on the network. The basic encoding rules of ASN.1 are used.

ASN.1

Standards ISO — 8824 and ISO — X.680.

<http://www.oss.com/asn1/>[Accessed 30/10/08]

<http://asn1.elibel.tm.fr/en/index.htm>[Accessed 30/10/08]

<http://asn1.elibel.tm.fr/en/standards/index.htm>[Accessed 30/10/08]

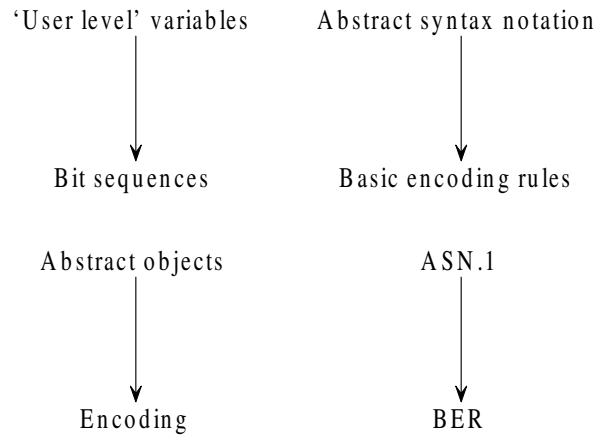
<http://www.asn1.org/>[Accessed 30/10/08]

Used to define:

- abstract syntaxes of application data
- structure of application and presentation protocol data units (PDUs)
- the management information base for SNMP and OSI systems management.

Terms

- Abstract syntax
 - Describes the generic structure of data independent of any encoding technique used to represent the data. Allows data types to be defined and values of those types to be specified.
- Data type
 - A named set of values. Simple (set of values specified) or structured (defined in terms of other types)



Need to recognise 4 kinds of tokens

- Words, letters, digits '-' (starting with a letter)
 - All uppercase = keyword
 - Initial letter uppercase = type (user defined)
 - Others = variables
- Numbers
- Strings
 - “character strings”
 - ‘0123456789abcdef’H
 - ‘01’B
- Punctuation
 - ::= - assignments
 - -- - comments

Data Types

- BOOLEAN ; INTEGER; ENUMERATED; REAL; BIT STRING; OCTET STRING
- Subsets of OCTET STRING
 - NumericString ; PrintableString ; IA5String ; CharacterString (+ others)
- GeneralizedTime
- UTCTime
- OBJECT IDENTIFIER
- NULL
- EXTERNAL
- ANY

Constructed Data Types

- SEQUENCE / SEQUENCE OF
- SET / SET OF
- CHOICE

Tags and Classes of Data Types

- Tags - labels for encoding purposes
- Four classes of tags
 - Universal
 - * The basic data types defined above
 - Application
 - * Defined within specific ASN.1 module
 - Context-Specific
 - * Tags for constructed data types
 - Private
 - * Enterprise-specific — no external co-ordination

Values

- Used to define type + instance
 - Not concerned with encoding
- BOOLEAN — TRUE / FALSE
- INTEGER — Numeric digit
- Strings (various) — use quotes
- Constructed data types
 - Recursively give each field's type, value

Modules

- A collection of ASN.1 definitions, syntax :

```
\modulename DEFINITIONS ::=
```

```
BEGIN
```

```
    \externalreferences
```

```
    \declarations
```

```
END
```

- Where

```
\modulename has two elements
```

- Textual name

- Object identifier

\externalreferences define references imported / exported by the module

\declarations defines the data types used by this module

Example 1

```
Month ::= INTEGER (1..12)
```

```
Day ::= INTEGER (1..31)
```

```
PositiveInteger ::= INTEGER (0<..PLUS-INFINITY)
```

```
personnelRecord ::= SEQUENCE{  
    empNumber INTEGER,  
    name IA5String,  
    yrsWithCompany INTEGER,  
    married BOOLEAN}
```

Example 2

```
Rectangle ::= SEQUENCE{  
    width Measurement,  
    length Measurement }
```

```
Measurement ::= SEQUENCE {  
    yards INTEGER,  
    feet INTEGER,  
    inches REAL }
```

```
Measurement ::= SEQUENCE {  
    yards INTEGER,  
    feet INTEGER (0 .. 2),  
    inches INTEGER (0 .. 1199) }  
-- value of inches in hundredths of an inch
```

SMI language constructs(rfc1155)

- OBJECT-TYPE — this construct specifies the data type, status and semantics of a managed object.
- SYNTAX — this clause of an OBJECT-TYPE specifies the data type associated with the object.
- ACCESS — this clause specifies whether the object can be read, written or is not accessible.
- STATUS — this clause specifies whether the object is mandatory, optional or obsolete.
- DESCRIPTION — this clause describes the managed object.

```
RFC1213-MIB DEFINITIONS ::= BEGIN
```

```
    IMPORTS
```

```
        mgmt, NetworkAddress, IpAddress, Counter, Gauge,
            TimeTicks
        FROM RFC1155-SMI
    OBJECT-TYPE
        FROM RFC-1212;
```

```
-- This MIB module uses the extended OBJECT-TYPE macro as
-- defined in [14];
```

```
-- MIB-II (same prefix as MIB-I)
```

```
mib-2      OBJECT IDENTIFIER ::= { mgmt 1 }
```

```
-- textual conventions
```

```
DisplayString ::=
```

```
    OCTET STRING
```

```
-- This data type is used to model textual information taken
-- from the NVT ASCII character set.  By convention, objects
-- with this syntax are declared as having SIZE (0..255)
```

```
PhysAddress ::=
```

```
    OCTET STRING
```

```
-- This data type is used to model media addresses.  For many
-- types of media, this will be in a binary representation.
-- For example, an ethernet address would be represented as
-- a string of 6 octets.
```

```

-- groups in MIB-II

system      OBJECT IDENTIFIER ::= { mib-2 1 }

interfaces  OBJECT IDENTIFIER ::= { mib-2 2 }

at          OBJECT IDENTIFIER ::= { mib-2 3 }

ip          OBJECT IDENTIFIER ::= { mib-2 4 }

icmp        OBJECT IDENTIFIER ::= { mib-2 5 }

tcp         OBJECT IDENTIFIER ::= { mib-2 6 }

udp         OBJECT IDENTIFIER ::= { mib-2 7 }

egp         OBJECT IDENTIFIER ::= { mib-2 8 }

-- historical (some say hysterical)
-- cmot      OBJECT IDENTIFIER ::= { mib-2 9 }

transmission OBJECT IDENTIFIER ::= { mib-2 10 }

snmp        OBJECT IDENTIFIER ::= { mib-2 11 }


-- the System group

-- Implementation of the System group is mandatory for all
-- systems.  If an agent is not configured to have a value
-- for any of these variables, a string of length 0 is
-- returned.

sysDescr OBJECT-TYPE
    SYNTAX  DisplayString (SIZE (0..255))
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "A textual description of the entity.  This value
        should include the full name and version
        identification of the system's hardware type,
        software operating-system, and networking
        software.  It is mandatory that this only contain
        printable ASCII characters."
    ::= { system 1 }

```

sysObjectID OBJECT-TYPE

SYNTAX OBJECT IDENTIFIER

ACCESS read-only

STATUS mandatory

DESCRIPTION

"The vendor's authoritative identification of the network management subsystem contained in the entity. This value is allocated within the SMI enterprises subtree (1.3.6.1.4.1) and provides an easy and unambiguous means for determining 'what kind of box' is being managed. For example, if vendor 'Flintstones, Inc.' was assigned the subtree 1.3.6.1.4.1.4242, it could assign the identifier 1.3.6.1.4.1.4242.1.1 to its 'Fred Router'."

::= { system 2 }

sysUpTime OBJECT-TYPE

SYNTAX TimeTicks

ACCESS read-only

STATUS mandatory

DESCRIPTION

"The time (in hundredths of a second) since the network management portion of the system was last re-initialized."

::= { system 3 }

sysContact OBJECT-TYPE

SYNTAX DisplayString (SIZE (0..255))

ACCESS read-write

STATUS mandatory

DESCRIPTION

"The textual identification of the contact person for this managed node, together with information on how to contact this person."

::= { system 4 }